ABDUCTIVE REASONING DRIVEN APPROACH TO PROJECT-LIKE PRODUCTION FLOW PROTOTYPING

G. Bocewicz*, Z Banaszak*

*Department of Computer Science and Management, Technical University of Koszalin, 75-453 Koszalin, Poland, zbigniew.banaszak@tu.koszalin.pl

Abstract: Constraint Programming (CP) is an emergent software technology for declarative description and effective solving of large combinatorial problems especially in the area of integrated production planning. In that context, CP can be considered as an appropriate framework for development of decision making software supporting scheduling of multi-robot in a multi-product job shop. The paper deals with multi-resource problem in which more than one shared renewable and non-renewable resource type may be required by manufacturing operation and the availability of each type is time-windows limited. The problem belongs to a class of NP-complete ones. The aim of the paper is to present a knowledge based and CLP-driven approach to multi-robot task allocation providing a prompt service to a set of routine queries stated both in straight and reverse way. Provided examples illustrate both cases while taking into account an accurate as well as an uncertain specification of robots and workers operation time.

Keywords: knowledge engineering, modeling, constraints logic programming, scheduling

1. INTRODUCTION

Some industrial processes simultaneously produce different products using the same production resources. An optimal assignment of available resources to production steps in a multi-product job shop is often economically indispensable. The goal is to generate a plan/schedule of production orders for a given period of time while minimizing the cost that is equivalent to maximization of profit. In that context executives want to know how much a particular production order will cost, what resources are needed, what resources allocation can guarantee due time production order completion, and so on. So, a manager’s needs might be formulated in a form of standard, routine questions, such as: Does the production order can be completed before an arbitrary given deadline? What is the production completion time following assumed robots operation time? Is it possible to undertake a new production order under given (constrained in time) resources availability while guaranteeing disturbance-free execution of the already executed orders? What values and of what variables guarantee the production order will completed following assumed set of performance indexes?

The problems standing behind of the quoted questions belong to the class of so called project scheduling ones. In turn, project scheduling can be defined as the process of allocating scarce resources to activities over a period of time to perform a set of activities in a way taking into account a given performance measure. Such problems belong to NP-complete ones. Therefore, the new methods and techniques addressing the impact of real-life constraints on the decision making is of great importance, especially for interactive and task oriented DSSs designing [4], [8].

Several techniques have been proposed in the past fifty years, including MILP, Branch-and-Bound [6] or more recently Artificial Intelligence. The last sort of techniques concentrates mostly on fuzzy set theory and constraint programming frameworks. Constraint Programming/Constraint Logic Programming (CP/CLP) languages [6], [20] seems to be well suited for modeling of real-life and day-to-day decision-making processes in an enterprise [5].
In turn, applications of fuzzy set theory in production management [21] show that most of the research on project scheduling has been focused on **fuzzy PERT** and **fuzzy CPM** [12], [13]. In this context, the contribution provides the framework allowing one to take into account both: distinct (pointed), and imprecise (fuzzy) data, in a unified way and treated in a unified form of a discrete, constraint satisfaction problem (**CSP**) [4]. The approach proposed concerns logic-algebraic method (**LAM**) based and **CP**-driven methodology aimed at interactive decision making based on distinct and imprecise data. The paper can be seen as continuation of our former works concerning projects portfolio prototyping [5], [11].

The following two classes of standard routine queries are usually considered and they are formulated in:

**a straight way** (i.e. corresponding to the question: What results from premises?)
- What the portfolio makespan follows from the given project constraints specified by activity duration times, resources amount and their allocation to projects’ activities?
- Does a given resources allocation guarantee the production orders makespan do not exceed the given deadline?
- Does the projects portfolio can be completed before an arbitrary given deadline?

**a reverse way** (i.e. corresponding to the question: What implies conclusion?)
- What activity duration times and resources amount guarantee the given production orders portfolio makespan do not exceed the deadline?
- Does there exist resources allocation such that production orders makespan do not exceed the deadline?
- Does there exist a set of activities’ operation times guaranteeing a given projects portfolio completion time will not exceed the assumed deadline?

Above mentioned categories encompass the different reasoning perspectives, i.e. deductive and abductive ones. The corresponding queries can be stated in different models that in turn may be treated as compositions of variables and constraints, i.e. assumed sets of variables and constraints limiting their values. In that context both an enterprise and the relevant production orders can be specified in terms of distinct and/or imprecise variables, discrete and/or continuous variables, renewable and/or non-renewable resources, limited and/or unlimited resources, and so on.

Possible problems formulation taking into account commercially available software packages capabilities is shown in the Table 1. So, that is easy to observe that commercially available tools are not able to consider cases assuming imprecise data as well as are not able to state a problem in an reverse way (e.g., looking for values of some input variables guaranteeing the assumed output variables reach required values). Moreover, the commercially available DSSs are not able to respond in an interactive, i.e. on-line/real-time mode, as well as to support a project-like production flow prototyping (i.e. integrated production planning containing such partial problems as routing, batch-sizing and scheduling).

That disadvantage is our motivation to develop methodology supporting one in the course of designing of an interactive and task oriented decision support systems aimed at projects portfolio prototyping. By projects prototyping we mean a decision process resulting in selection (variables adjustment) both an enterprise and projects portfolio parameters fulfilling assumed requirements, e.g. an admissible solution being a kind of an equilibrium between enterprise capabilities and projects’ cost and makespan.